**Project Report**

The project report is outlined in six sequential steps comprising the CRISP-DM framework.

1. **Business Understanding**The business wants a binary classification system that can correctly predict if a customer (of that business) is likely to buy a car a second time. This will help them arrange business and marketing campaigns around the system as they can now base their decisions on the classification model, thereby making the most out of their efforts.

The resources available is one single dataset in csv format with around 130 000 entries. This meets the project requirement as the dataset is large enough to train and test Machine Learning models well. However, there are several risks involved. Since business strategies will depend a lot on our model’s decisions, the stakes could be high in terms of cost. However, on the flip side, the project’s benefits would be worth it since it would automate in crucial decision making which normal human beings would take longer to do.

The success of this project would be to identify the best classification model that predicts the likelihood of re-purchase of a customer. Not only would it have high accuracy, but it would be reproducible and reliable for future datasets to come.

The project being run would largely make use of Python to clean and prepare data, then build and test machine learning models. Following this it will also help us create data visualizations to help us find insights and sort out findings.

1. **Data Understanding**

In our first step, we collect initial data. We import the dataset into our coding environment and work on the data.

The dataset contains 131, 337 entries and 16 rows in total. Among the columns, we have 4 of them that are not numerical; they are text but are categorical data. However, the age\_band column is in a numerical range format.

The first column is the ID column – one containing all unique values. Thus it has no relationship with other columns whatsoever. The numeric columns seem to fall in a very narrow numeric range – 0 to 10, and their statistical descriptions are very similar. For example, the mean, standard deviation and even the inter-quartile ranges for all the columns are same for some reason.

Digging a little deeper tells us that there are missing values in the age\_band and gender column. After calculation, it is found out that 85% of the age\_band and 52% of the gender are missing. However, other than that the dataset is pretty clean, there are no duplicate values, no outliers and no discernable noise, which makes data preparation easier.

1. **Data Preparation**

Because of the high missing values in the age\_band and gender column, it is tough to judge how they can be filled. One way to consider would be to replace the values with their statistical mode, but doing so imposes the risk of assumption which is 85% and 52% for the two columns. Thus, the models we would create could be misdirected by “assumed” values. Another way of fixing the problems could be imputations, but we do not have a strong base of data to do that. Thus these two columns will be removed from our study. Now it is important to note that age and gender are two valuable parameters for our scenario, from a practical standpoint. However, this may be one of the limitations of the project that we are excluding two highly valuable parameters.

The ID column is also dropped out because it provides no numerical significance for our machine learning models.

The category value 'car\_model' is present in the dataset. The problem is that higher models correspond to newer brands if we treat it as an ordinal and assume that the model names refer to some kind of chronology. Alternately, we can think of it as a nominal, in which case the model names cannot be arranged in any particular order. Because it's possible that the model\_15 and model\_16, for instance, were released for that car brand in the same year, I've taken it as nominal. Because of this, we are unable to arrange models 15 and 16 in any sort of sequential order.

Thus, one-hot encoding was applied to the column, producing 19 extra columns (or features) of boolean values.

Following this, the car\_segment column was mapped in the following manner:   
"Other": 0, "LCV": 1, “Small/Medium": 2, "Large/SUV": 3

Then the final dataset was created by merging these columns. The original car\_model column was removed since it was no longer needed.

Finally, the new dataset was scaled with a standard scaler to have all the values fit within the same range.

1. **Modelling**

There were 5 experiments done with 4 models to run. They were Logistic Regression, SVM, Decision Trees and Random Forests. In our 5th experiment, we optimized hyperparameters using Grid Search and Random Search.

Before the model was run, the data was split into 3 sets – train, validation and test. We then ran the model by instantiating the class of that model and then fit the model with the dataset.

Now we analyzed the success criteria of our models by assessing the accuracy scores, f1 scores, Precision and recall. We have displayed confusion matrices as well to visualize the scores properly.

1. **Evaluation**

Interestingly enough, all the four models experimented with did a great job in meeting the business criteria. Although some of them had overfitting issues, their accuracy scores were pretty high nonetheless, along with Precision and f1. (Recall was down for some models, but it was still not a problem considering business context because not detecting false negatives is not much of a problem for the business).

Having said that, it was found that SVM was the best-suited model for the business. It had an f1 score of 0.95, scored high on Precision and Recall and it did not face any overfitting issues.

There was a step missing in the project though – we did not apply any algorithms for optimizing the hyperparameters of SVM. Maybe if we applied the grid search, we can know which parameters to tune to get the best results. However, those parameters are not guaranteed to work best for future unknown datasets.

Thus based on all these analyses and observations, we can decide to deploy the SVM model for the businesss.

1. **Deployment**

For deploying the model, the SVM can be containerized and then deployed. But the main work is once the deployment is done. The model needs to go through thorough monitoring and maintenance to avoid issues. For example it needs to be ensured that it is not going through garbage data, otherwise it will become skewed and start giving erroneous results.

Also, as a project team we need to document the result of our model and project it to the business as per the requirements.

In retrospection, having the two columns (age\_band and Gender) that we missed out on would be really crucial if we want to improve on this project further. This would give us a more grounded conclusion on customer behavior and vehicle re-purchase